4Н

УДК 681.31

Я.М. Николайчук, О.І. Волинський, С.В. Кулина
Тернопільський національний економічний університет, м. Тернопіль, Україна

Швидкодіючий алгоритм та процесор порівняння чисел у системі залишкових класів базису Крестенсона

У статті досліджено можливості СЗК для побудови алгоритму порівняння чисел, а також створення схемотехнічної моделі повнофункціонального 16-розрядного процесора.

Вступ

В наш час одним з перспективних шляхів розвитку комп'ютерної теорії є вивчення непозиційних систем числення, а саме системи залишкових класів (СЗК), яка базується на використанні базису Крестенсона. Одним з основних завдань в комп'ютерній техніці є створення повнофункціональних процесорів. Тому незалежно від того, яка з систем розглядається – позиційна чи непозиційна, основним поштовхом для її розвитку є прагнення створити повнофункціональний процесор чи спецпроцесор, який зможе виконати поставлені перед ним задачі.

У даній статті розглянуто можливість створення алгоритмічної моделі виконання операції порівняння та проектування процесора в НДСЗК.

1. Теоретичні основи СЗК

1.1. Теоретичні основи цілочисельної форми представлення СЗК

В основу цілочисельного перетворення СЗК покладена Китайська теорема про залишки [1]. Суть прямого перетворення цілочисельної форми СЗК полягає в тому, що згідно з теоремами про залишки будь-яке ціле число може однозначно перетворити набором найменших невід'ємних залишків в системі взаємопростих модулів, що відповідає рішенню діафантового рівняння

$$N_k = b_i (\mathrm{mod}\, p_i),$$

яке відповідає цілочисельному рішенню лінійного рівняння:

$$N_k = a_i p_i + b_i ,$$

де a_i – ранг, b_i – найменший невід'ємний залишок.

При цьому діапазон кодування чисел N_k дорівнює:

$$P = \prod_{i=1}^{k} p_i \text{ ; } 0 \le N_k \le P.$$

Таким чином ціле число N_k однозначно представляється набором залишків b_i.

Зворотне перетворення цілочисельної форми СЗК виконується згідно з аналітичним виразом

$$N_k = res\sum_{i=1}^{k} b_i \cdot B_i (\mathrm{mod}\ P), \qquad (1)$$

де B_i – базисні числа СЗК, які обчислюються згідно з діафантовим рівнянням:

$$B_i = \frac{P}{p_i} \cdot m_i \equiv 1(\mathrm{mod}\ \ P_i)\,. \qquad (2)$$

1.2. Теоретичні основи нормалізованої форми представлення СЗК

Теоретичною основою утворення нормалізованої форми СЗК (НСЗК) [2] є нормалізація по модулю Р обох частин рівняння зворотнього перетворення цілочисельної форми СЗК:

$$\frac{N_k}{P} = res\sum_{i=1}^{k} \frac{b_i \cdot B_i(\mathrm{mod}\,P)}{P}, \qquad (3)$$

звідки

$$[N_k]_0 = res\sum_{i=1}^{k} b_i \cdot \frac{B_i}{P}(\mathrm{mod}\,1),$$

а з врахуванням виразу (2) отримаємо:

$$[N_k]_0 = res\sum_{i=1}^{k} b_i \cdot \frac{m_i}{p_i}(\mathrm{mod}\,1) \text{ або } [N_k]_0 = res\sum_{i=1}^{k} [b_i]_0 \cdot m_i(\mathrm{mod}\,1), \qquad (4)$$

де $[b_i]_0 = \dfrac{b_i}{p_i}$, а $0 \le [b_i]_0 \le 1$.

Для забезпечення однозначного кодування даних в НСЗК необхідно виконувати умову:

$$\delta_p \le \frac{1}{P}\,.$$

Дана формула визначає необхідне число розрядів після коми, у відповідній системі числення при представлені величини 1/Р в нормалізованій формі, тобто

$$\frac{1}{p_i} = 0.\overbrace{gggg}^{n_i}\overbrace{gggg}^{\delta_p},$$

де g – цифри у відповідній системі числення, n_i – число розрядів, до яких заокруглюється результат ділення з приведенням до меншого цілого, а δ_p – дробова частина, яка визначає величину похибки δ_p, якою нехтують.

Таким чином аналітичний вираз з НСЗК в СЗК отримує вигляд:

$$N_k = \mathrm{int}[N_k]_0 \cdot P,$$

де int – символ операції виділення цілої частини.

Формула перетворення СЗК (1) може бути представлена в наступному вигляді:

$$N_k = res\sum_{i=1}^{k} b_i \cdot \frac{P}{p_i} \cdot m_i(\mathrm{mod}\,P), \qquad (5)$$

де $0 \le m_i \le p_i - 1$.

Очевидно, що наявність коефіцієнтів m_i в формулі (4) ускладнює реалізацію алгоритму виконання цілочисельного перетворення СЗК. Дослідження різних набо-

рів pі, яким відповідають набори коефіцієнтів m_i в теоретико-числовому аспекті показали, що існують такі набори модулів p_1, p_2, ..., p_k, які відповідають умовам взаємної простоти з одиничними коефіцієнтами $m_i(m_1=m_2=...=m_i=...=m_k=...=1)$.

Прикладом такого набору модулів є $p_1=2$, $p_2=3$, $p_3=5$. Для якого P=30, $B_1=15$, $B_2=10$, $B_3=6$, а $m_1=m_2=m_3=1$.

Перевагою НСЗК є виконання операцій над залишками в нормалізованій формі, що спрощує реалізацію процесорів на основі даного базису, за рахунок виключення нелінійних операцій отримання залишку по кожному з модулів процесора, а також заміни операції по «mod P» на операцію по «mod 1», яка виконується шляхом простого відкидання цілої частини результату згідно з операцією int.

1.3. Теоретичні основи нормалізованої досконалої форми представлення СЗК

Теоретичною основою даної форми є рівняння (3), підставивши в яке $m_1=m_2=m_k=1$, оттримаємо базове рівняння перетворення НДСЗК у вигляді:

$$[N_k]_0 = res\sum_{i=1}^{k}[b_i]_0(\text{mod}1). \qquad (6)$$

З рівняння (6) видно, що з перетворення НДСЗК виключена операція множення і саме перетворення виконується у вигляді сумування нормалізованих залишків $[b_i]_0$ по mod 1, що відповідає операції int відкидання цілої частини результату.

При дослідженні наборів модулів, які відповідають нормалізованій досконалій формі СЗК було з'ясовано, що ці модулі не є оптимальними для реалізації процесорів певної розрядності. Оскільки довільному набору взаємопростих модулів відповідає довільний набір коефіцієнтів m_i, це обмежує можливості створення процесорів з оптимальним набором модулів.

Нами запропонований алгоритм приведення будь-якої цілочисельної форми СЗК до НДСЗК, який реалізується шляхом корекції графа визначення залишків згідно з розрахованим набором коефіцієнтів m_i [3].

Для реалізації 16-бітного процесора оптимальний набір модулів з максимальним діапазоном кодування задається масивом чисел:

$$p_i = (3, 5, 7, 8, 11, 13).$$

Для даної СЗК:

Діапазон кодування: $P = 3\cdot5\cdot7\cdot11\cdot13 = 120120$.

Базисні числа:

$$B_1 = 80080, m1 = 2;$$
$$B_2 = 96096, m2 = 4;$$
$$B_3 = 85800, m3 = 5;$$
$$B_4 = 105105, m4 = 7;$$
$$B_5 = 76440, m5 = 7;$$
$$B_6 = 36960, m6 = 4.$$

Скоректований граф обчислення залишків згідно з розрахованим набором коефіцієнтів m_i буде мати вигляд:

$B_1 = 3$	$B_2 = 5$	$B_3 = 7$	$B_4 = 8$	$B_5 = 11$	$B_6 = 13$
0	0	0	0	0	0
0,6666 +	0,8 +	0,7142857 +	0,875 +	0,6363636 +	0,3076923≡ (mod 1)0,0000083
0,3333	0,6	0,428571	0,75	0,272727	0,615384
	0,4	0,142857	0,625	0,909091	0,923076
	0,2	0,8571428	0,5	0,545454	0,230769
		0,571428	0,375	0,181818	0,538461
		0,285714	0,25	0,818181	0,846153
			0,125	0,454545	0,153846
				0,090909	0,461538
				0,727272	0,769230
				0,363636	0,076923
					0,384615
					0,692307

Рисунок 1 – Скоректований граф обчислення залишків

Зі схеми обчислення числа за графом (рис. 1) зрозуміло, що це значно спрощує перехід в базис Радемахера і виконання операції порівняння чисел.

2. Реалізація алгоритму та сопроцесора порівняння чисел на основі НДСЗК

2.1. Алгоритм порівняння чисел в НДСЗК

Для виконання операції порівняння двох чисел в НДСЗК потрібно звернутися до рис. 1. В пам'яті ЕОМ граф обчислення залишків буде відтворений в другій системі числення (рис. 2).

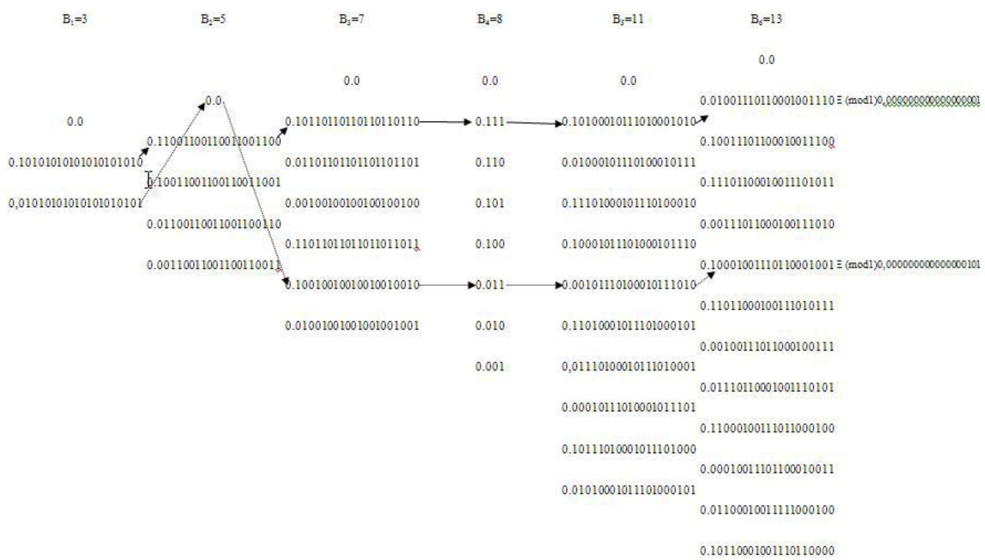


Рисунок 2 – Двійкове відображення графа обчислення залишків

Для того, щоб виконати операцію порівняння двох нормалізованих чисел в НДСЗК, потрібно виконати операцію сумування нормалізованих залишків цих чисел згідно з рис. 2. Підставивши у формулу (6) двійкові коди залишків, взяті з рис. 2 отримаємо формули:

$$[N_i]_{(0,2)} = res\sum_{i=1}^{n}[b_i]_{(0,2)}(\mathrm{mod}1)_{(2)}\ , \qquad [N_j]_{(0,2)} = res\sum_{i=1}^{n}[a_i]_{(0,2)}(\mathrm{mod}1)_{(2)}\ .$$

Для реалізації алгоритму ділення необхідно виконати операцію порівняння $[N_i]_0$ і $[N_j]_0$.

2.1. Алгоритм порівняння чисел в НДСЗК

Для реалізації операції порівняння запропоновано наступну структуру спецпроцесора:

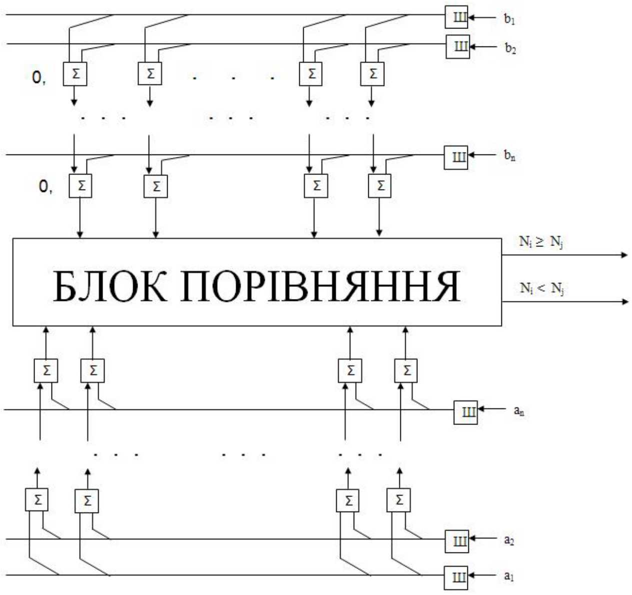


Рисунок 3 – Структура спецпроцесора для виконання операції порівняння в базисі Крестенсона

Висновок

В роботі запропоновано використання нормалізованої досконалої системи залишкових класів для реалізації швидкодіючого перетворення з СЗК в нормалізовану двійкову систему запису Радемахера, що дозволяє виконати операцію порівняння двох чисел у базисі Крестенсона на базі запропонованої структури спецпроцесора. Це значно спрощує реалізацію операції ділення в СЗК і створює умови для побудови повнофункціонального процесора СЗК.

Література

1. Акушский И.Я., Юдицкий Д.И. Машинная арифметика в остаточных класах. – М: Сов. радио, 1968. – 440 с.
2. Николайчук Я.М., Федорович Ю.С. Теоретичні основи базисних перетворень СЗК // Матеріали наукової конференції «Автоматика 2000». – Львів. – 2000. – С. 120.
3. Николайчук Я.М., Волинський О.І., Кулина С.В. Теорія побудови та компоненти швидкодіючих процесорів на основі досконалої та розмежованої форм системи залишкових класів // Матеріали проблемно-наукової міжгалузевої конференції «Інформаційні проблеми комп'ютерних систем, моделювання, економіки та юриспруденції» (SPIS 2008). – Збірник наукових праць Бучацького інституту менеджменту і аудиту. – 2008 – Т. 1. – Випуск № 4. – С. 31-35.

Я.М. Николайчук, О.И. Волынский, С.В. Кулина

Быстродействующий алгоритм и процессор сравнения чисел в системе остаточных классов базиса Крестенсона

В статье исследованы возможности системы остаточных классов для создания алгоритма сравнения чисел, а также создание схемотехнической модели полного функционального 16-разрядного процессора.

Стаття надійшла до редакції 17.07.2008